(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
10 May 2002 (10.05.2002)

PCT

(10) International Publication Number
WO 02/37328 A2

(51) International Patent Classification⁷: G06F 17/30

(21) International Application Number: PCT/IL01/00942

(22) International Filing Date: 11 October 2001 (11.10.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
09/690,307 17 October 2000 (17.10.2000) US

(71) Applicant *(for all designated States except US)*: **FOCUSENGINE SOFTWARE LTD.** [IL/IL]; 12 Raul Valenberg Street, 69719 Tel Aviv (IL).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **DAGAN, Ido** [IL/IL]; Shivtei Israel Street 26A, 47267 Ramat Hasharon (IL). **FUKS, Avi** [IL/IL]; 6 Shilo Street, 64688 Tel Aviv (IL). **PAVLOVITZ, Ofra** [IL/IL]; Sokolov Street 6, 52571 Ramat Gan (IL). **YELLIN, Ido** [IL/IL]; Jerusalem Street 8/2, 44446 Kfar Saba (IL).

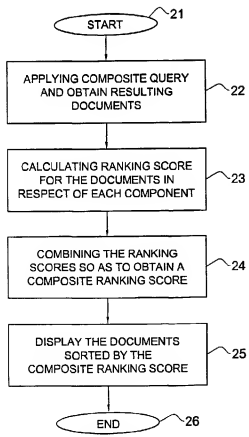
(74) Agent: **REINHOLD COHN AND PARTNERS**; P.O. Box 4060, 61040 Tel Aviv (IL).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PH, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian

[Continued on next page]

(54) **Title:** INTEGRATING SEARCH AND CLASSIFICATION: SCORING AND RANKING



(57) **Abstract:** A system for obtaining a composite score of documents that includes a user interface for providing a composite query that includes a free-text query component and a category component and obtain documents that meet the composite query. The system further includes a processor for calculating a non-Boolean score of the document according to each one of the components. The processor is further configured to combine the scores so as to obtain a composite score, displaying through the user interface the documents associated with said score, possibly sorted by the scores.

WO 02/37328 A2

patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

INTEGRATING SEARCH AND CLASSIFICATION: SCORING AND RANKING

FIELD AND BACKGROUND OF THE INVENTION:

The amount of textual information that is available in computerized media has increased dramatically in recent years. As a result, there is an increasing need for end users to have effective tools for searching, browsing, navigating, reading and analyzing collections of textual documents. Current common practice, within organizations as well as in the Internet, is having a search engine that indexes a large repository of documents and enables users to issue a search query and to get in response all documents that satisfy the search conditions.

Usually, a list of titles, along with some additional information, is presented for each document and the user can further ask for the display of specific documents from the list. The list of documents is often sorted by some relevance ranking, which is intended to approximate the degree of relevance of the document to the query. Sorting by date is also often available.

A search mechanism typically attaches to each document a set of *indexing concepts*. An indexing concept is a symbol or value that characterizes the document, and is typically used within search queries or within routing queries ("queries" that specify which documents will be routed to an addressee). Typical types of indexing concepts include topical categories (also known as controlled keywords, topics, descriptors etc.). These are symbols denoting topical issues, which are usually general or abstract concepts that do not necessarily appear literally in the text. For example, a topical category may be "Company

Acquisition". This term, serving as the name of the category, may not appear literally in a document that describes such an event.

In the following, a document is considered *indexed* by the indexing concepts characterizing it. Apart from being used in ad-hoc search queries, indexing concepts may also be used to determine routine routing of incoming documents to addressees.

The process of associating indexing concepts to documents (the *indexing* process) is performed either manually, automatically, or by some combination of the two modes. With respect to indexing concepts that consist of terms and names from the document text, the indexing process usually involves scanning the text of the document, identifying words, terms and names, and possibly bringing these terms to some canonical form (e.g. the grammatical base form (lemma) of the word).

Of particular interest is the indexing process for topical categories (*categories*, in short). In many systems, it is possible for the user to manually assign topical categories to a document. More recently, there have been developed a number of methods for assigning topical categories to documents automatically, which are referred to here as automatic text classification methods. Such methods classify documents to appropriate categories taken from a predetermined list of possible categories. Classification is performed by some mechanism that receives the document text as input and determines the appropriate categories based on the words, terms or their combinations that appear in the document. The mechanism scores every document in relation to every category, and a document is classified to a category if its score is above some predefined threshold.

There are two common approaches for automatic text classification methods. The first approach is based on manual definition of the rules, or some other type of logic, by which a document is being

classified to a category based on the terms in the text. For example, some systems allow users (or administrators) to define complex queries, which may include Boolean and other types of conditions (such as weights and proximity) that the terms in the document should satisfy. A document that satisfies these conditions is classified to the category. An example for such a system is the Topics ™ system that was developed by Verity Inc., USA. Typically, the characterization of a category is referred to as the "profile" of the category. Basically, the profile is a weighted vector of terms, but it can include more sophisticated conditions as described above. Every document is scored according to the correlation between the profile and the terms that appear in it. The second approach is based on automatic learning of the "logic" which entails the classification of the document to a category. Methods belonging to this approach utilize a set of *training* documents, for which the correct categories are known in advance (usually as the result of manual classification of these documents). A learning method may then include a learning phase, in which some model of the category is constructed. For example, such a model may include terms that are highly associated with the category, and possibly some weights that quantify the degree of correlation (entailment) between each term and the category. Alternatively, a learning method may be *memory based*, in which case the learning method simply stores the training data in some useful format. Then, when a new document is given for classification, the method classifies it automatically by consulting or applying the category model (or by simply comparing the document to the training data, in case of a memory based approach). Examples for trainable (learning) classification systems are described in:

1. C. Apte and F. Damerau and S. Weiss, 1994. Towards language independent automated learning of text categorization models, in Proceedings of ACM-SIGIR Conference on Information Retrieval.

2. W.W. Cohen, Text categorization and relational learning, in Machine Learning Journal, 1995, pages 124—132.

3. W. W. Cohen and Y. Singer, Context-sensitive learning methods for text categorization, in Proceedings of the 19th Annual Int. ACM Conference on Research and Development in Information Retrieval, 1996, pages 307—315.

4. D. Lewis, 1992, An evaluation of phrasal and clustered representations on a text categorization problem, in Proc. of the 15th Int. ACM-SIGIR Conference on Information Retrieval, pages 37—50.

5. D. Lewis and M. Ringuette, 1994, A comparison of two learning algorithms for text categorization, in Proc. of Symposium on Document Analysis and Information Retrieval, pages 81—93.

6. D. Lewis and R. E. Schapire and J. P. Callan and R. Papka, 1996, Training algorithms for linear text classifiers, in SIGIR '96: Proc. of the 19th Int. Conference on Research and Development in Information Retrieval.

7. K. Tzeras and S. Hartmann, 1993, Automatic Indexing Based on Bayesian Inference Networks, in Proc. of 16th Int. ACM

SIGIR Conference on Research and Development in Information Retrieval, pages22—34.

8. E. Wiener and J. Pedersen and A. Weigend, 1995, A neural network approach to topic spotting, in Symposium on Document Analysis and Information Retrieval, pages 317—332.

Once documents have been obtained by a user, as a result of some search or some routing mechanism, these documents are typically displayed in one of several formats and ranked according to their relevance.

Any form of display that takes document scores into account is often sorted by some relevance ranking, which is intended to approximate the degree of relevance of the document to the query. Typically the query includes some free-text terms and the scoring is dependent on various known *per se* criteria such as the number, frequency and positioning of the free text terms in the document.

Users often use composite queries, that include both a basic component (e.g. free-text component like "laser") and a category component like "science". In these cases there is a need to take categories into account in the scoring process. There is known in the art a degenerated form of taking in account also categories in the scoring process. However, in this degenerated form the categories are taken in account only in a Boolean manner. For a better understanding of the foregoing, consider the following example, illustrating the operation in accordance with hitherto known techniques in the following search engine:

http://hotbot.lycos.com/

if one searches the HOTBOT DIRECTORY, selecting, say "science" , a list of various sub-categories is obtained, see http://dir.hotbot.lycos.com/Science/

Then, one can either click on any category from among these sub-categories or start a search within the category "science".
For example, if the word "laser" (standing for the query) is used for the "Search this Category " option then one gets:
http://hotbot.lycos.com/?MT=laser&RT=OD&CID=1j1337&Search.x=39&Search.y=10

In other words, the query "laser" is searched within the category "science".

When one searched in the category (science), the resulting documents were scored only by the free-text query (laser). The only effect of the category was that only documents that belong to the category were shown (i.e. documents that meet the query "laser" but which belong to a category other than "science" are not shown.). In other words, scoring the documents by category is Boolean.

Considering that the categories also reflect an interest of the user (e.g. in the latter example not only the term "laser" is of interest but also the category "science") there is a need in the art to reflect in the scoring results of the documents the effect of the "category" in a more fine-tuned manner rather than the hitherto known coarse Boolean scoring criterion.

There is a further need in the art to combine at least (i) the resulting category score with (ii) conventional scoring of the document according to query, bringing about a composite score of the document.

SUMMARY OF THE INVENTION:

In accordance with the invention the scoring of the so retrieved documents takes into account score of categories in addition to other scores. The latter include the basic query score (such as the free-text

words that were introduced to the query), but as will be explained in greater detail below possibly also other known *per se* scoring criteria.

It should be noted that in the context of the invention a query is not bound to any free text form and accordingly any form of query that produces a set of results with scores is applicable (hereinafter basic query). One example of a basic query is a free-text query. Other options such as browsing a directory or asking for similar pages are also applicable.

It should be further noted that the term document should be construed in a broad manner including, but not limited to, text documents represented in various formats, multimedia documents that include audio and/or video.

It should be further noted that following the scoring phase where a documents are assigned with composite scores, there follows a display step where the documents (or data associated therewith such as titles) are displayed, preferably according to some ranking criterion. In accordance with a non-limiting example the ranking is realized by sorting the documents by their composite scores and displaying all (or some of them according to a pre-defined criteria), in, say descending composite score order.

It should be further noted that the invention is not bound by any particular interface for placing the query(s) or obtaining the query results, and accordingly the appropriate interface may be determined, depending upon the particular application.

It should be further noted that in accordance with the invention depending upon the particular application the term category encompasses both pre-determined categories and ad-hoc categories. Accordingly, the score a document is given in relation to a category may be the result of a supervised classification (into pre-determined categories, using some

automatic classification method) or an unsupervised classification (into ad-hoc categories, using some clustering algorithm).

In accordance with a preferred embodiment of the invention, a composite query is composed of at least a basic query component (e.g. free-text query component) and indexing concept component and more specifically category component. It should be noted that each of the said components may comprise several sub-components: the basic query component may be a free-text phrase that includes several words; similarly, the indexing concept may include several categories. Each document has a composite score for the query as a whole. This score is determined by scores for each of the query components, both the free-text component and the categories (which by themselves may be the result of combining the scores for their sub-components) which are then composed so as to obtain a composite score of the document.

Thus in accordance with the invention there is provided a method for obtaining a composite score of documents, comprising:

i) providing a composite query that includes at least basic query component and indexing concept component and obtain at least one document that meet said composite query;

ii) calculating a non-Boolean score of said at least one document according to each one of said components;

iii) combining said scores so as to obtain a composite score; and

iv) displaying at least one of said documents associated with said score.

The invention further provides a system for obtaining a composite score of documents, comprising:

i) means that include user interface for providing a composite query that includes at least basic query component and indexing concept component and obtain at least one document that meet said composite query;

ii) means that include processor for calculating a non-Boolean score of said at least one document according to each one of said components;

iii) means that include processor combining said scores so as to obtain a composite score; and

iv) means that include user interface for displaying at least one of said documents associated with said score.

In accordance with a preferred embodiment, combining the scores is accomplished by taking into account relationships between the components within the document, such as adjacency.

In accordance with a preferred embodiment a filtering condition is applied to the score of the query so as to consider only documents that match the query at a score that meets the specified filtering criterion. By a specific embodiment this filtering criterion being a threshold and only those documents whose score exceed the specified threshold are considered for the subsequent category score and the scoring combination step (which bring about the composite score of the document. It should be noted that for convenience of explanation the term composite score is referred to occasionally in short as score).

In accordance with a preferred embodiment and will be explained in greater detail below, category score of the document is not only combined with query score of the specified document but possibly also with other scores of the documents, e.g. the date of the document. In other words, other factors which are not necessarily related to the

specified query/category components may be weighted and combined to the composite score.

Thus the invention further provides a method for obtaining a composite score of documents, comprising:

i) providing a composite query that includes at least indexing concept component that is constituted by at least two sub-components and obtain at least one document that meet said composite query;

ii) calculating a non-Boolean score of said at least one document according to each one of said components;

iii) combining said scores so as to obtain a composite score; and

iv) displaying at least one of said documents associated with said score.

Still further the invention provides a system for obtaining a composite score of documents, comprising:

i) means that include user interface for providing a composite query that includes at least indexing concept component that is constituted by at least two sub-components and obtain at least one document that meets said composite query;

ii) means that include processor for calculating a non-Boolean score of said at least one document according to each one of said components;

iii) means that include processor combining said scores so as to obtain a composite score; and means that include user interface for displaying at least one of said documents associated with said score.

In accordance with another embodiment of the invention the use of basic query component is obviated. Thus, for example, a composite query is composed only of an indexing concept component (e.g., that includes several categories), in which case the composite score is determined by combining the scores of the distinct category sub-components. The various modifications discussed above apply also to this embodiment. For example, other score (such as date) may be utilized in constructing the composite score.

BRIEF DESCRIPTION OF THE DRAWINGS:

For a better understanding of the foregoing the invention will now be described by way of example only with reference to the accompanying drawings, in which:

Fig. 1 is a generalized schematic illustration of a system in accordance with an embodiment of the invention;

Fig. 2 is a flow chart illustrating a generalized sequence of operation in accordance with a preferred embodiment of the invention; and

Figs. 3A-B illustrate screen results according to hitherto known database search system which will assist in clarifying a category scoring step that is utilized in the system and method of the invention.

DESCRIPTION OF PREFERRED EMBODIMENTS:

It should be noted that for convenience of explanation the description below referrers to a free-text query component. As explained above, free text query component is only out of many possible variants of basic query component.

Attention is now drawn to Fig. 1 illustrating a generalized schematic system (**10**) in accordance with an embodiment of the

invention. As shown, plurality of user nodes (by this example nodes **11, 12** and **13**) communicate through communication medium **(14),** e.g. the Internet with a server **(15).** The user nodes running e.g. a browser application and place a query that consists, e.g. of plurality of free-text key words and possibly some categories. The query is processed wholly at server **(15)** (or divided among the user node and the server node) and the resulting documents and their associated composite score is displayed at the user node screen. The server hold(s) database of documents and/or other documents repository.

It should be noted that the invention is by no means bound by the schematic architecture illustrated in Fig. 1.

Thus, by way of non-limiting examples: in accordance with a modified embodiment, other network(s) may be utilized in addition or instead of the Internet. In accordance with another modified embodiment, the query is applied locally not through a communication network. In accordance with yet another modified embodiment, more than one server is utilized. In accordance with another modified embodiment, any user nodes may include one of the following: personal computer, Personal Digital Assistant (PDA), or Cellular telephone, Other variants are applicable all as required and appropriate.

Attention is now directed to Figs. 3A-B which will assist in understanding the sequence of operation in accordance with a preferred embodiment of the invention.

Thus, U.S. patent 5,924,090 (Krellenstein) "Method and Apparatus for Searching a Database of Records" discloses system for searching a database and present to the user a small number of categories along with a list of most relevant documents that satisfy a query. The methodology of the Krellenstein patent has a sophisticated clustering algorithm that includes three primary steps: identifying candidate

categories, weighting candidate categories and displaying a set of search result categories selected from the candidate categories.

A typical result of the system according to the Krellenstein patent is illustrated in Figs. 3A-B, as extracted from the www.northernlight.com site. Thus, as shown the free-text component of the query *"text categorization"* **(31)** results in 19,215 documents (records) **(32)** (of which 6 are shown in the first page). The documents are assigned to 15 categories **(33).** The set of categories are determined after applying the specified sophisticated clustering including identifying candidate categories, weighting candidate categories (so as to obtain categories score) and displaying a set of search result categories selected from the candidate categories. As specified above the selection depends, of course, upon the so calculated score. It goes without saying that due to the coarse "Boolean" criterion that is used in the technique according to the Krellenstein patent, some categories (such as sport) are displayed notwithstanding the fact that they have low or no relevance.

In accordance with the specified system, the user can repeat this process further narrowing the search with each iteration. Thus, double clicking the category *"Special collection documents"* **(34)** will result in applying the specified steps again giving rise to the search results illustrated in Fig. 3A. It should be noted that the category *"Special collection documents"* stands for the category component of the query and accordingly the composite query includes by this example a free-text component *"text categorization"* and category component *"Special collection documents"*. As shown there are 2057 documents **(35)** in the sought category **(36)** that, in turn are assigned to 12 categories **(37).**

It should be noted that in the specified prior art system the score of free-text component is non-Boolean (e.g. score that ranges over a fine tuned scale, as known *per se*) and the score of the category component

is Boolean. Insofar as the latter is concerned this constitutes a significant shortcomings. Thus, a document is displayed in the specified category if it belongs thereto and is not displayed if it does not belong thereto. There is no indication as to "to what extent" the document belongs to the category or "to what extent" it does not belong to the specified category. Put differently there is no Non-Boolean score for the categories and *a fortiori* there is no combination between the respective non-Boolean scores of the free text component and the category component.

Before turning to Fig. 2, it should be noted that the various elements described in Fig. 2 may be implemented in the user and the server nodes, depending upon the particular application. Thus, in accordance with a non–limiting example the calculating and combining steps are realized at the remote server site.

Bearing this in mind, attention is now drawn to Fig. 2 illustrating a flow chart of a generalized sequence of operation in accordance with a preferred embodiment of the invention. As a first stage a composite query is applied to the database (and/or any other document repository) **(22)** similar to the composite query with a free-text component "*text categorization*" and category component "*Special collection documents*" discussed above. As shown the composite query is not necessarily applied in one step and, if desired, may be constructed in several stages. For example in the latter embodiment the free text component is applied as a first step and thereafter the category component is designated. The process may be continued iteratively by designating additional free-text components and category components.

Having obtained the resulting documents that meet the query, the documents are scored in respect of each component **(23)**. The free-text score aims at determining how relevant the key words are to the document and there are numerous scoring techniques that may be

employed to this end e.g. in accordance with the conventional search engines such as *Alta Vista*™ search engine where each document is associated with a non-Boolean score, signifying how relevant is the document to the free-text query words. The higher the score the more relevant is the document.

In accordance with the invention, a non-Boolean score is calculated in respect of the category component. For example, the score for the category component may be the one obtained by applying some non-supervised classification algorithm such as e.g. in accordance with the specified Krellenstein Patent. However, unlike the hitherto known techniques where the non-Boolean score is mapped to a Boolean value (belong or does not belong to the category), in accordance with a preferred embodiment of the invention the fine tuned score is maintained and utilized in the next step. By another preferred embodiment a supervised algorithm such as using profiles for classifying to categories may be utilized.

Thus, in the next step **(24)** a composite score is determined by some mechanism that combines the scores of the distinct scored components. By way of non limiting example the composite score takes into account relationships between the matches of the components in the document, say any one or combination of the following operators: sum, product, average, weighted average, geometric mean, or minimum of the component scores. Insofar as the latter example is concerned, there may be various considerations what operator or operators to employ. By way of non limiting example *geometric mean* is preferable over *average* if the composite score should emphasize a significant contribution of every component and not only one of them. Consider for example the following simplified scenarios: in accordance with a first scenario the score of the free-text component is 8 and of the category is 2 and a

second scenario where the score of the free-text component is 5 and of the category is 5. Whereas the average in both scenarios is 5, the geometrical mean is 4 and 5 respectively. Thus, should it be desired to emphasize the "contribution" of both components (i.e. each contributing "5" in the second scenario as compared to significant contribution of only one component "8" in the first scenario), one should select the geometrical mean as a composite score operator (giving rise to a composite scores 5 vs. 4) rather than the average operator (which in both scenarios resulted in composite score 5).

Obviously, more than one operator and/or other operators may be employed, depending upon the particular application.

The combination step may employ not only "mathematical" (mathematical encompasses also "logical") operators, e.g. of the kind specified. Thus, in accordance with a modified embodiment other operators are employed in addition or in lieu the specified mathematical operators. For example, order of components in the query may be taken in account, where e.g. the later the component the more weight it receives. By a modified embodiment certain components *a priori* receive more weight, say the free-text component benefits from higher weight than the category component etc.

In accordance with another modified embodiment, the combination step utilizes in addition or in lieu of the specified operators proximity/distance operators, one example being the adjacency operator. Thus, in accordance with one variant of the specified modified embodiment each paragraph is scored by the number of different matches in it. In accordance with this embodiment a "bonus" is conferred to the overall score as a function of the number of paragraphs with much intersection between the query components. Consider the above referred to example where the free text component is "laser" and

the category component is "science". If the elements in the text that "contribute" to the score of the free text component "laser" and those that contribute to the score of the category "science" (e.g. the term in the category's profile) reside in the same paragraph it means that the specified paragraph or paragraphs of the document are related to laser and science (which was the initial contemplation of the query issuer) and accordingly a higher composite score should be achieved. In contrast, a lower composite score should be conferred in a scenario where, say, the terms that contribute to "laser" reside in one paragraph (attesting that this paragraph is indeed related to "laser") and the terms that contribute to "science" reside in another separate paragraph (attesting that this paragraph is indeed related to "science"). Whilst the latter document indeed "discusses" *laser* and *science* it does not necessarily discuss laser in a scientific context (which was the original contemplation of the query issuer). Thus, for example, the first paragraph may discuss "laser pointer" and the second (separated) paragraph may discuss "scientific matters" which do not concern lasers.

By another modified embodiment the adjacency operator also takes into account the "weight" of the matching profile or free-text query term. That is, terms in the profile and in the query may have strength (profile weight, general term weight in the query – like the known per se Inverted Document Frequency – IDF). The boost entailed by adjacent query and profile terms should be larger if these are terms with high weight.

In the case that the free-text component and the category components are scored in different scales it is required to apply a normalization step in order to bring the respective scores to comparable scales, or weighted in order to allow for comparable effects of the score

components. Another possibility is some empirical normalization to bring the scores to the same scale.

Those versed in the art will readily appreciate that the invention is not bound in the specified mathematical and non-mathematical operators in the score combination step.

Whereas the description above focused predominantly in free-text query component and category query component, in accordance with another modified embodiment additional components may be utilized. Thus, by way of non-limiting example the Google™ incorporates factors related to the number and quality of links pointing at a document. The specified component may be combined in the composite score e.g. by adding "bonus score" in the case of qualitative links. In accordance with another modified embodiment the document's date may also be a factor, where, say, new documents receives a bonus score as compared to older document. Other modified components may be utilized in addition or in lieu of the above, all as required and appropriate.

Having obtained composite score, the documents are displayed along with their associated score. By one embodiment the documents are sorted, ranked and displayed (e.g. as a whole or title or abstract, all as known *per se)* according to the composite score, say in a descending order. By way of another example, the documents are displayed in a hierarchy of categories, according to their classification by some classification algorithm.

Standard search engines present all matches of a free-text query in a list ordered by match score. Thus there's no need to set a threshold of a minimal score, since the user sees only the first part of the list and can see the rest upon request.

In certain embodiments of the invention where the resulting documents are displayed in hierarchical form it may be necessary to set

such a threshold. Consider the following scenario: some documents may have low scores for the composite query, but they are the only documents in some category (note that the query isn't necessarily a free-text query, it might be any combination of free-text queries and category selection operations). In that case, the category appears in the hierarchy but when the user "drills down" into the category, the documents found there are actually of very low relevance for the query.

To fix this situation, a threshold is set for the minimal score (in the free-text component) a document should have in order to be displayed in the hierarchy. Thus, this category will not be displayed at all. For other categories the threshold may imply a lower number of documents within the category.

By another preferred embodiment of the invention only indexing concepts (e.g. categories) form the query (e.g. the query might be "the category "science" *and* the category "news", resulting in documents that are classified to both these categories) and accordingly the composite (non-Boolean) score is based only on scores of indexing concepts. If desired other factors may be utilized in order to give rise to composite score, such as date and/or order, all as explained in detail above.

It will also be understood that the system according to the invention may be a suitably programmed computer. Likewise, the invention contemplates a computer program being readable by a computer for executing the method of the invention. The invention further contemplates a machine-readable memory tangibly embodying a program of instructions executable by the machine for executing the method of the invention.

In the following alphabetic character and roman symbols are used for convenience only and do not necessarily imply any particular order of the method steps.

The present invention has been described with a certain degree of particularity but those versed in the art will readily appreciate that various alterations and modifications may be carried out without departing from the scope of the following claims:

CLAIMS:

1) A method for obtaining a composite score of documents, comprising:

i) providing a composite query that includes at least basic query component and indexing concept component and obtain at least one document that meet said composite query;

ii) calculating a non-Boolean score of said at least one document according to each one of said components;

iii) combining said scores so as to obtain a composite score; and

iv) displaying at least one of said documents associated with said score.

2) The method according to Claim 1, wherein the calculation of the non-Boolean indexing concept component score includes applying non-supervised algorithm to terms in the documents so as to determine said indexing concept score.

3) The method according to Claim 2, wherein said non-supervised algorithm being in accordance with the Krellenstein technique.

4) The method according to Claim 1, wherein the calculation of the non-Boolean indexing concept component score includes applying a supervised algorithm to terms in the documents so as to determine said indexing concept score.

5) The method according to Claim 4, wherein said supervised algorithm includes mapping to indexing concepts according to profiles.

6) The method according to any one of the preceding claims, wherein said index concept is a category.

7) The method according to any one of the preceding claims, wherein said basic query component being free-text component.

8) The method according to any one of the preceding claims, wherein said combining step includes applying one or more mathematical operator.

9) The method according to Claim 8, wherein said mathematical operator is selected from the group that includes: sum, product, average, weighted average, geometric mean, or minimum.

10) The method according to any one of Claims 1 to 7, wherein said combining step includes applying one or more non–mathematical operator.

11) The method according to Claim 10, wherein said combining step includes applying one or more non–mathematical operator.

12) The method according to Claim 10, wherein said non–mathematical operator being proximity/distance operator.

13) The method according to any one of the preceding claims, wherein said composite query includes at least one additional component and wherein said method further comprising the step of:

calculating at least one additional score of said at least one document in respect of at least one additional component; and combining said additional score in said step (c) so as to obtain said composite score.

14) The method according to Claim 13, wherein said additional component being the document date.

15) The method according to Claim 13, wherein said additional component being the order of said basic query component and category component in said composite query, such that additional score is assigned to the additional component depending on the order thereof.

16) The method according to any one of the preceding claims, further comprising ranking at least one of said documents according to its composite score and displaying said at least one document according to its rank

17) The method according to any one of the preceding claims, wherein said combining step includes a preliminary normalization step in order to bring said components to comparable scale.

18) The method according to any one of the preceding claims, further comprising applying a threshold on the basic component score before combining it with the indexing concept score.

19) A method for obtaining a composite score of documents, comprising:

i) providing a composite query that includes at least indexing concept component that is constituted by at least two sub-components and obtain at least one document that meets said composite query;

ii) calculating a non-Boolean score of said at least one document according to each one of said components;

iii) combining said scores so as to obtain a composite score; and

iv) displaying at least one of said documents associated with said score.

20) The method according to claim 19, wherein said composite query includes at least one additional component and wherein said method further comprising the step of:

calculating at least one additional score of said at least one document in respect of at least one additional component; and combining said additional score in said step (c) so as to obtain said composite score.

21) A system for obtaining a composite score of documents, comprising:

i) means that include user interface for providing a composite query that includes at least basic query component and indexing concept component and obtain at least one document that meet said composite query;

ii)means that include processor for calculating a non-Boolean score of said at least one document according to each one of said components;

iii) means that include processor combining said scores so as to obtain a composite score; and

iv) means that include user interface for displaying at least one of said documents associated with said score.

22) A system for obtaining a composite score of documents, comprising:

i) means that include user interface for providing a composite query that includes at least indexing concept component that is constituted by at least two sub-components and obtain at least one document that meet said composite query;

ii) means that include processor for calculating a non-Boolean score of said at least one document according to each one of said components;

iii) means that include processor combining said scores so as to obtain a composite score; and

iv) means that include user interface for displaying at least one of said documents associated with said score.

23) The system according to Claim 21, wherein said means are divided among client node and remote server node, communicating over communication network.

24) The system according to Claim 23, wherein said communication network being the Internet.

25) The system according to Claim 22, wherein said means are divided among client node and remote server node, communicating over communication network.

26) The system according to Claim 25, wherein said communication network being the Internet.

27) A Program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform method steps for obtaining a composite score of documents, comprising:

i) providing a composite query that includes at least basic query component and indexing concept component and obtain at least one document that meet said composite query;

ii)calculating a non-Boolean score of said at least one document according to each one of said components;

iii) combining said scores so as to obtain a composite score; and

iv) displaying at least one of said documents associated with said score.

28) A computer program product comprising computer useable media having computer readable program code embodied therein for obtaining a composite score of documents, the computer program product comprising:

computer readable program code for causing the computer to provide a composite query that includes at least basic query component and indexing concept component and obtain at least one document that meet said composite query;

computer readable program code for causing the computer to calculating a non-Boolean score of said at least one document according to each one of said components;

computer readable program code for causing the computer to combining said scores so as to obtain a composite score; and

computer readable program code for causing the computer to displaying at least one of said documents associated with said score.

29) A program storage device readable by machine, tangibly embodying program of instructions executable by the machine to perform method steps for obtaining a composite score of documents, comprising:

i) providing a composite query that includes at least indexing concept component that is constituted by at least two sub-components and obtain at least one document that meet said composite query;

ii) calculating a non-Boolean score of said at least one document according to each one of said components;

iii) combining said scores so as to obtain a composite score; and

displaying at least one of said documents associated with said score.

30) A computer program product comprising computer useable media having computer readable program code embodied therein for obtaining a composite score of documents, the computer program product comprising:

computer readable program code for causing the computer to provide a composite query that includes at least indexing concept component that is constituted by at least two sub-components and obtain at least one document that meet said composite query;

computer readable program code for causing the computer to calculating a non-Boolean score of said at least one document according to each one of said components;

computer readable program code for causing the computer to combining said scores so as to obtain a composite score; and

computer readable program code for causing the computer to displaying at least one of said documents associated with said score.

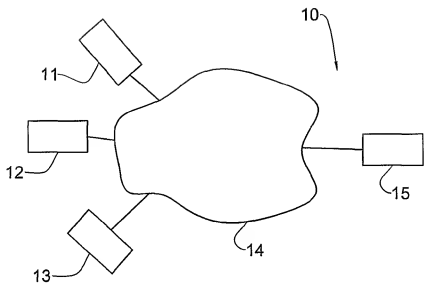

10
11
12
13
14
15

FIG. 1

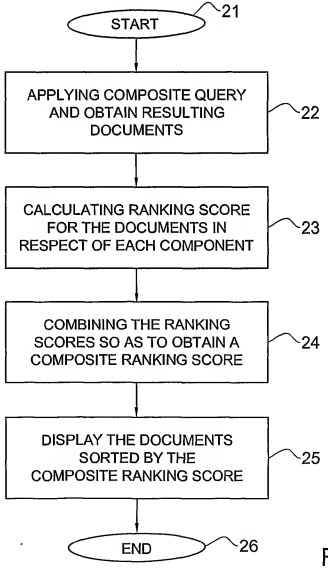

START
21
APPLYING COMPOSITE QUERY AND OBTAIN RESULTING DOCUMENTS
22
CALCULATING RANKING SCORE FOR THE DOCUMENTS IN RESPECT OF EACH COMPONENT
23
COMBINING THE RANKING SCORES SO AS TO OBTAIN A COMPOSITE RANKING SCORE
24
DISPLAY THE DOCUMENTS SORTED BY THE COMPOSITE RANKING SCORE
25
END
26

FIG. 2

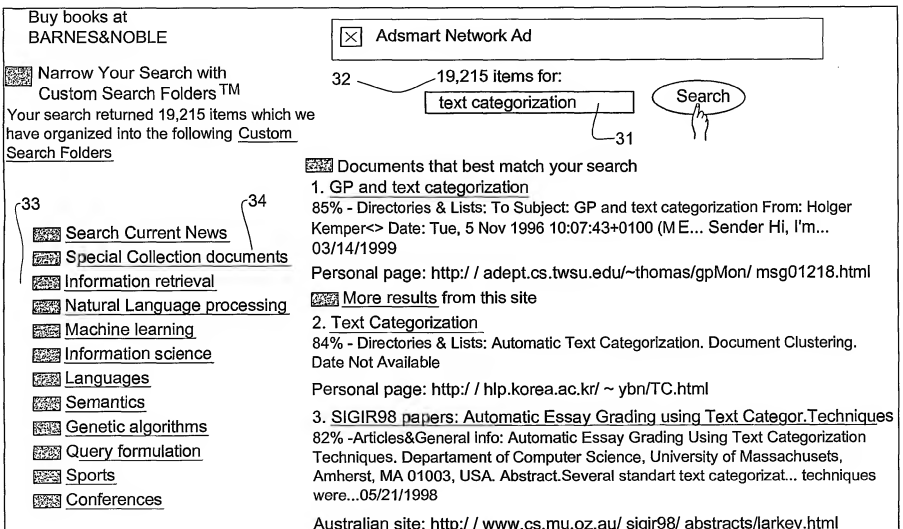

Buy books at
BARNES&NOBLE
Adsmart Network Ad
Narrow Your Search with
Custom Search Folders TM
32
19,215 items for:
text categorization
Search
31
Your search returned 19,215 items which we have organized into the following Custom Search Folders
Documents that best match your search
1. GP and text categorization
85% - Directories & Lists: To Subject: GP and text categorization From: Holger Kemper<> Date: Tue, 5 Nov 1996 10:07:43+0100 (ME... Sender Hi, I'm... 03/14/1999
Personal page: http:/ / adept.cs.twsu.edu/~thomas/gpMon/ msg01218.html
More results from this site
2. Text Categorization
84% - Directories & Lists: Automatic Text Categorization. Document Clustering. Date Not Available
Personal page: http:/ / hlp.korea.ac.kr/ ~ ybn/TC.html
3. SIGIR98 papers: Automatic Essay Grading using Text Categor.Techniques
82% -Articles&General Info: Automatic Essay Grading Using Text Categorization Techniques. Departament of Computer Science, University of Massachusets, Amherst, MA 01003, USA. Abstract.Several standart text categorizat... techniques were...05/21/1998
Australian site: http:/ / www.cs.mu.oz.au/ sigir98/ abstracts/larkey.html
33
34
Search Current News
Special Collection documents
Information retrieval
Natural Language processing
Machine learning
Information science
Languages
Semantics
Genetic algorithms
Query formulation
Sports
Conferences

FIG. 3A

Buy books at BARNES&NOBLE
Admart Network Ad
35
2,057 items in Special Collection documents for:
text categorization
Search
Narrow Your Search with Custom Search Folders TM
Your search returned 2,057 items which we have organized into the following Custom Search Folders
36
Special Collection documents
Nuclear waste
Health & medical services
Radiation&radiological protection
37
Law
Nuclear reactors
Books&readers' services
Artificial intelligence
Computer software products
Groupware
Non-fiction literature
Documents that best match your search
1. Special Collection Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorizati..
81% - Report summaries: Summary of report by: Carnegie-Mellon Univ., Pittsburgh, PA Dept. of Computer Science. 03/01/1996
National Technical Information Service - NTIS (report summaries): Available at Northern Light
More results from this publication
2. Special Collection: Text databases and information retrieval
66% - Articles & General Info: The goal of a traditional information retrieval (IR) system is to search an information repository, such as a text database, and retrieve documents that are...03/01/1996
ACM Computing Surveys (journal): Available at Northern Light
3. Special Collection: CONTENT CLASSIFICATION: Leveraging New Tools and Librarians Expertise.
56% -Articles&General info: A typical searcher enters a search term (or set of terms) into a search engine, searching a portal, whether an intranet, the Web, or online ...10/01/1999

FIG. 3B